(12) UK Patent Application (19) GB (11) 2 307 133 (13) A

(43) Date of A Publication 14.05.1997

(21) Application No 9523190.8

(22) Date of Filing 13.11.1995

(71) Applicant(s)
The Secretary of State for Defence

(Incorporated in the United Kingdom)

Defence Evaluation and Research Agency, DRA Farnborough, FARNBOROUGH, Hants, GU14 6TD, United Kingdom

(72) Inventor(s)
Robert William Maclaughlin Smith
Philip John Kent

(74) Agent and/or Address for Service
R W Beckham
D/IPR (DERA), Poplar 2, MOD (PE) Abbey Wood \19, PO Box 702, BRISTOL, BS12 7DU, United Kingdom

(51) INT CL^6
H04N 5/232

(52) UK CL (Edition O)
H4F FD12X FD24 FD30B FD30E FD30K FD31G FHHC

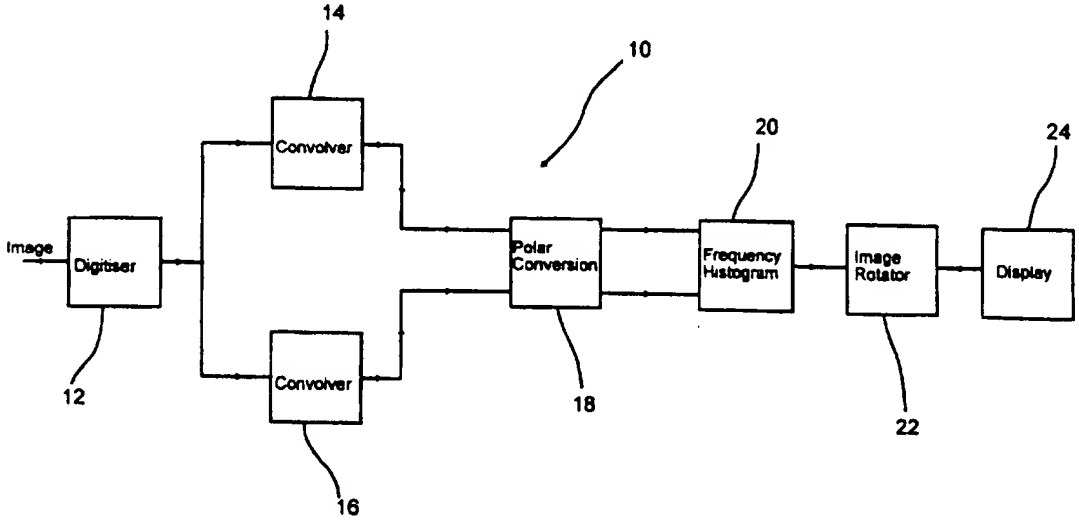
(56) Documents Cited
GB 2275842 A **GB 2074806 A** **EP 0079195 A2**
WO 90/00334 A1

(58) Field of Search
UK CL (Edition O) **H4F FESA FESG FESX FHHC**
INT CL^6 **F41G 3/22 7/22 , G02B 27/64 , H04N 5/232 5/262**
Online: WPI

(54) Video camera image stabilisation system

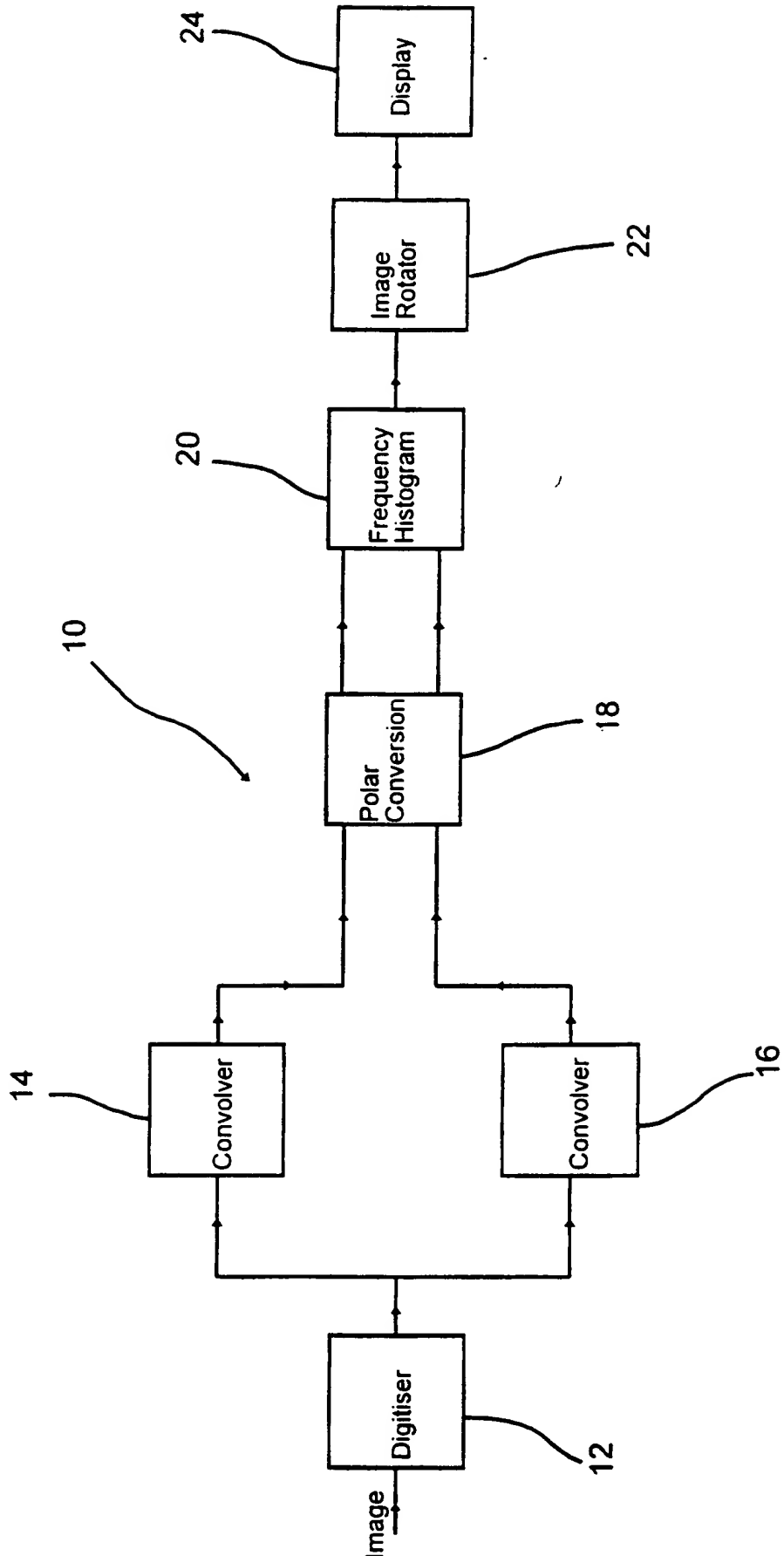
(57) An image stabilisation system provides means for counteracting a rotation of an image caused by a camera rotating about its optical axis. An image is digitised by a unit 12 and convolved with two filter kernels by two convolution units 14, 16 to enhance the vertical and horizontal components of edges in the image. A histogram of the orientations of the edges is then correlated with an edge orientation histogram of a previous image to determine a relative degree of rotation which is then used to correct the rotated image for display on a display unit 24.

FIGURE 1



GB 2 307 133 A

FIGURE 1



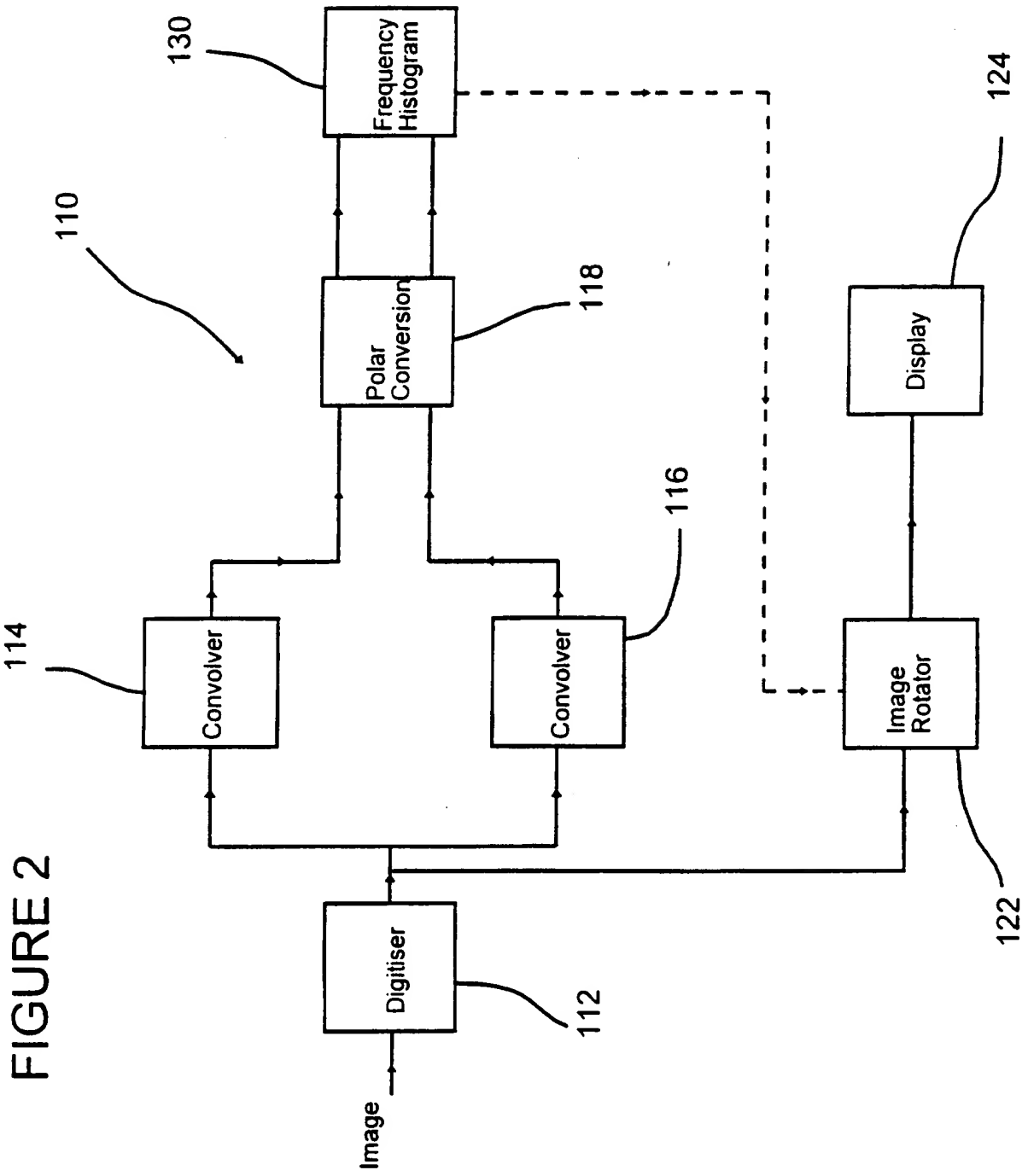


FIGURE 2

FIGURE 3

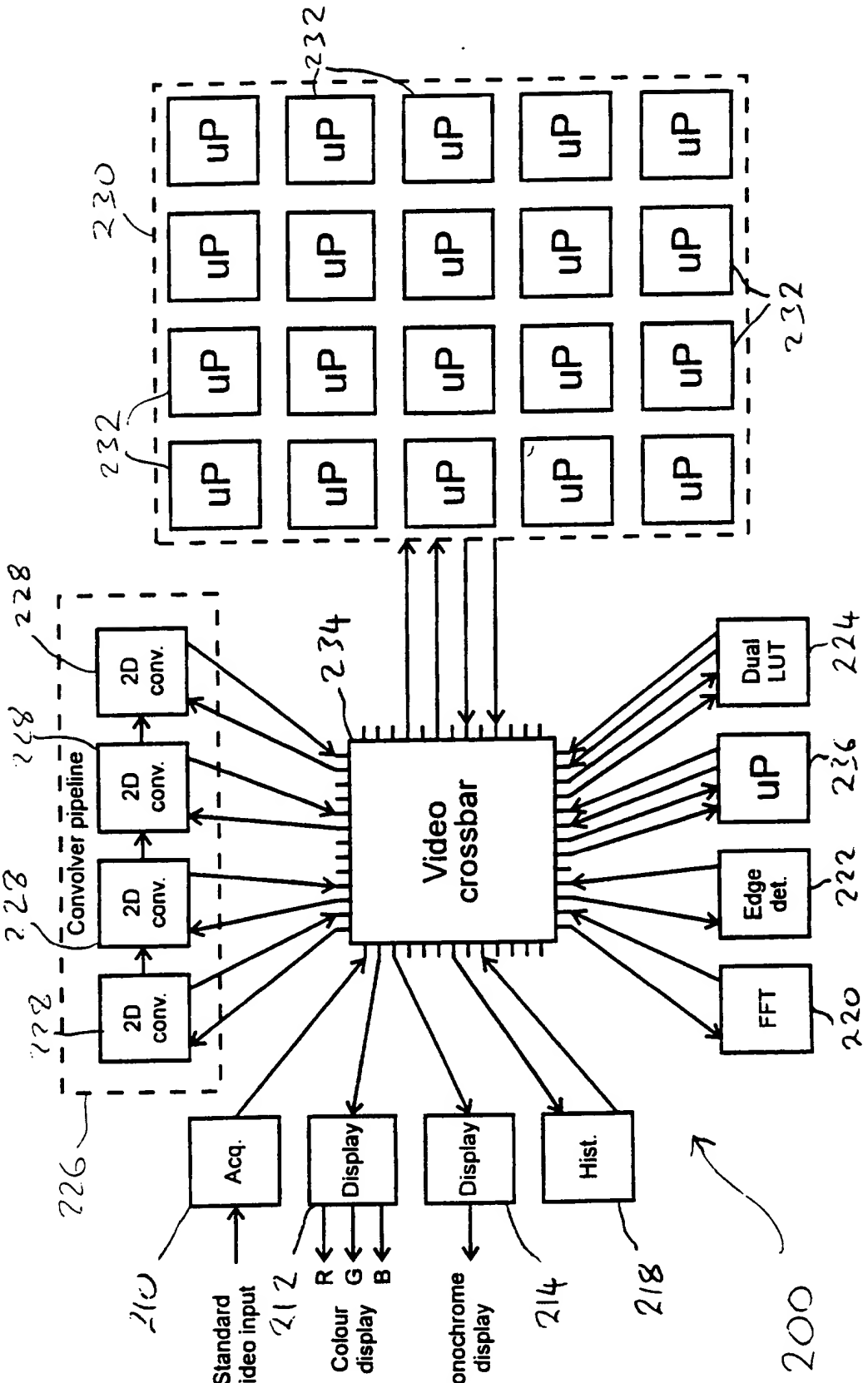


FIGURE 4

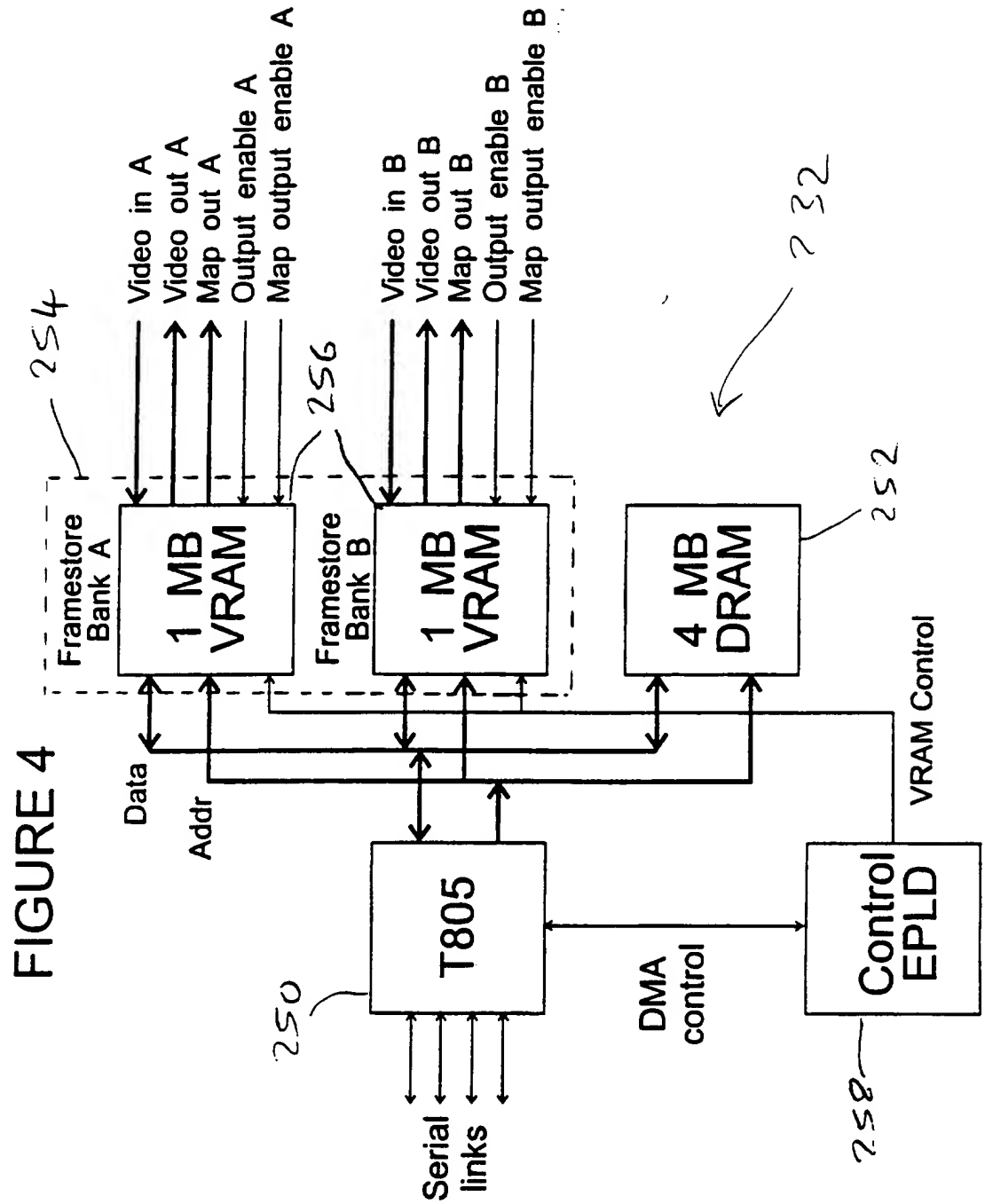


FIGURE 5

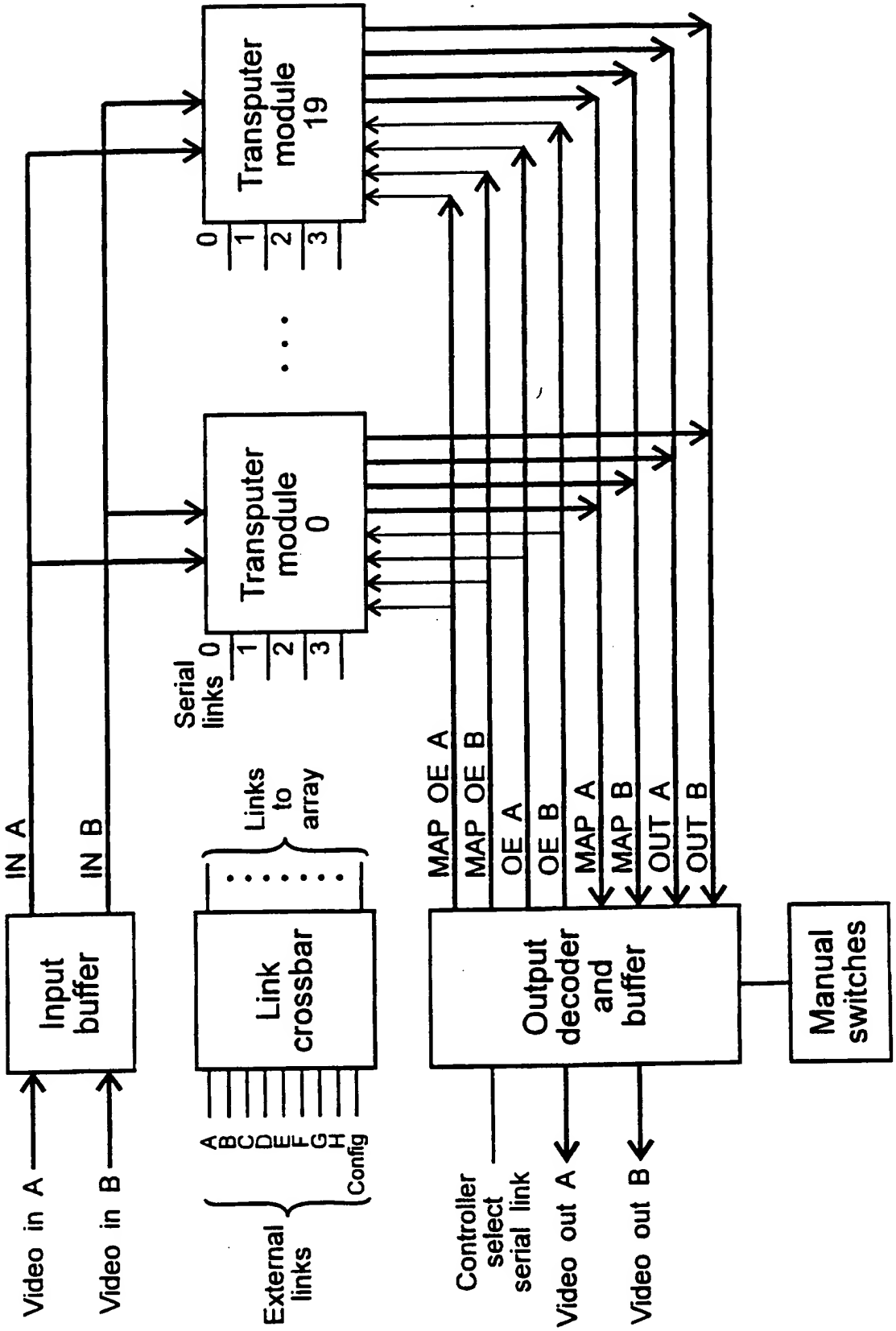


FIGURE 6

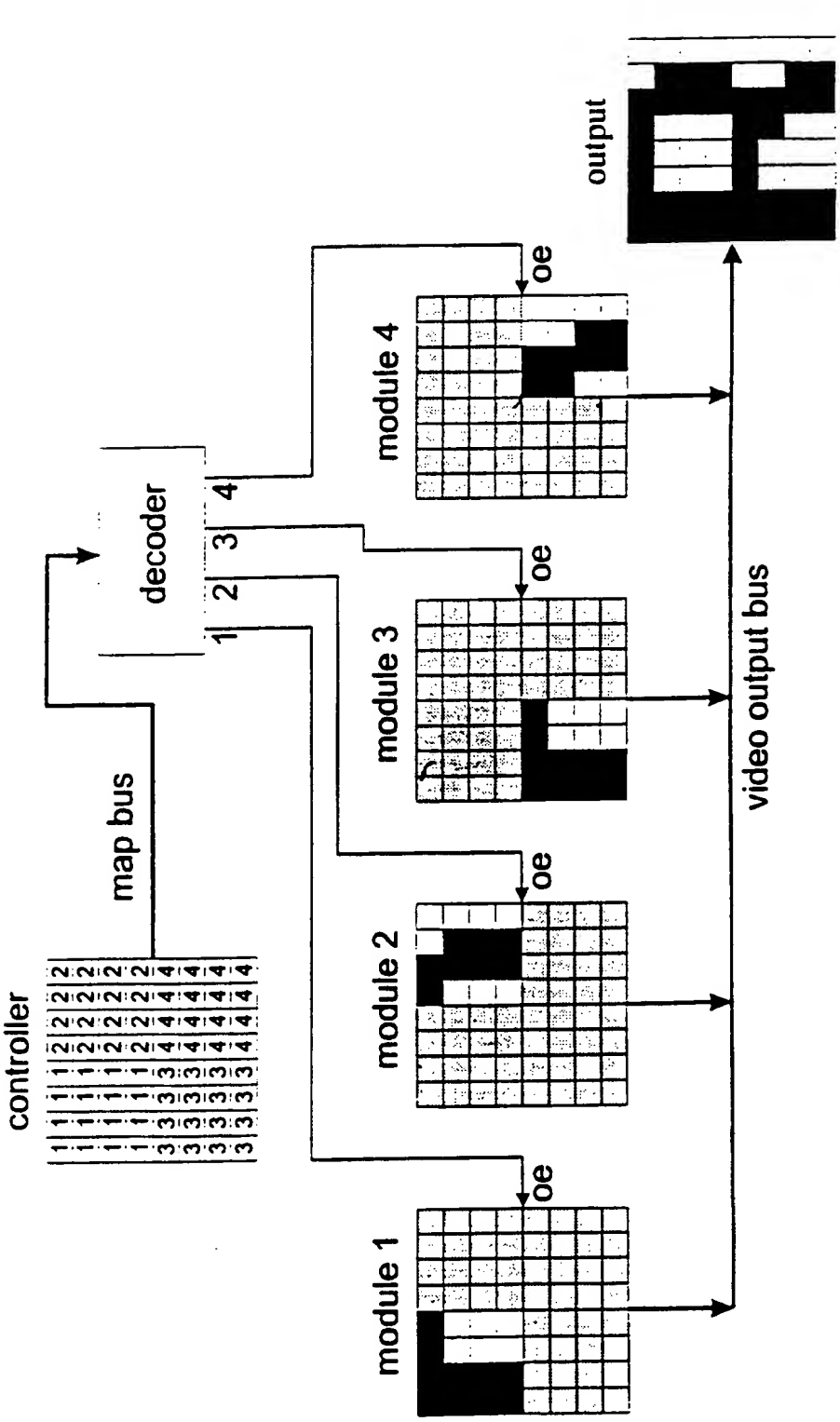


IMAGE STABILISATION SYSTEM

This invention relates to systems for stabilising images.

There are a number of applications in which an image is conveyed remotely from a moving camera to a viewing screen, such as in pipeline/borehole inspections or in remotely controlled undersea vehicles. However, movement of the vehicle or camera also causes the image on the viewing screen to move, and if this movement is rapid or rotational then information from the scene under observation could be lost.

Furthermore, if the viewing screen image is being used to control the vehicle or camera, then such movement may cause an operator to lose control with the possible result that the vehicle and/or camera may be damaged.

It is an object of the invention to provide an image stabilisation system.

The invention provides an image stabilisation system comprising means for determining an angular distribution of scene edges in an acquired image and means for correlating the edge angular distribution of the acquired image with the edge angular distribution of a previous image and for generating an alignment correction in response thereto.

The invention has the advantage that it enables images to be stabilised to correct for rotational movements of an image acquisition device, such as a rotating camera.

A further advantage of the invention is that the image is stabilised using only information contained within the image, without the need for additional sensor data, for example data from gyroscopes or other inertial systems.

The stabilisation may be performed without assuming the presence of any specific features in the image, for example the horizon or walls, which could be used to orientate the image. All that is required is that there are features in the scene which have non-circular edges in a grey-scale representation of the image.

In an alternative embodiment, the invention provides an image stabilisation system comprising means for determining an angular distribution of scene edges in an acquired image, means for correlating the edge angular distribution of the acquired image with the edge angular distribution of a previous image, means for predicting an alignment correction for the acquired image from an analysis of previous images, and means for determining an error term from a comparison of the predicted alignment correction and an actual alignment correction required to maximise the correlation between the angular distribution of the acquired image and the angular edge distribution of a previous image.

The invention further provides an image stabilisation system incorporating means for generating pixel image edge angles, wherein the rotational angle of this image is obtained from a comparison of the pixel image edges in a current image with the pixel image edges in a reference image.

The invention also provides a method of stabilising an image comprising the steps of
(i) determining an angular distribution of scene edges in an acquired image;
(ii) correlating the angular distribution of the edges in the acquired image with the edge angular distribution of a previous image; and
(iii) generating an alignment correction in response to the correlation.

The invention further provides a method of stabilising an image comprising the steps of
(i) determining an angular distribution of scene edges in an acquired image;
(ii) correlating the edge angular distribution of the acquired image with the edge angular distribution of a previous image;
(iii) predicting an alignment correction for the acquired image from an analysis of previous images; and
(iv) determining an error term from a comparison of the predicted alignment correction and an actual alignment correction required to maximise the correlation between the angular distribution of the acquired image and the angular edge distribution of a previous image.

Embodiments of the invention will now be described by way of example only with reference to the accompanying drawings in which:

Figure 1 shows a schematic diagram of an image stabilisation system;

Figure 2 shows a schematic diagram of an alternative image stabilisation system;

Figure 3 shows a functional-level diagram of an image processing system;

Figure 4 shows a diagram of a transputer module of the Figure 3 system;

Figure 5 shows a diagram of a transputer array of the Figure 3 system; and

Figure 6 shows an example of a video output of the Figure 5 transputer array.

Referring to Figure 1, there is shown an image stabilisation system, indicated generally by 10. The system 10 enables an image to be stabilised in situations where the image is captured by a camera (not shown) which is rotating about its optical axis. The system 10 incorporates a digitisation unit 12. The digitisation unit 12 receives an input from the camera and generates a digitised 256 x 256 pixel output image of an acquired image. The digitisation unit 12 is an electrical circuit which incorporates a KSV3110 analogue to digital converter and a SAA1043 video synchronisation circuit. The digitised image is input simultaneously to two convolver units 14 and 16. The convolver units 14 and 16 each comprises two 2D convolver devices, manufactured by SGS Thomson and having a product code IMSA110, cascaded together. The convolver units 14 and 16 are used to carry out vertical and horizontal edge-enhancement convolutions respectively. Such convolutions are described by L.Spacek in two documents - "Thinning Image Boundaries", CSM 187, Department of Computer Science, University of Essex, Colchester 1993; and "The Vision Library Manual", CSM 199, Department of Computer Science, University of Essex, Colchester 1993.

The convolver units 14 and 16 function as follows. Any edges in the image are found where the image has spatial intensity changes, and these are enhanced by convolving with two separate filter kernels which enhance the vertical and horizontal components of the edges of the images. The vertical component kernel is shown below:

5	8	10	8	5
2	5	10	5	2
0	0	0	0	0
-2	-5	-10	-5	-2
-5	-8	-10	-8	-8

and the horizontal component kernel is:

-5	-2	0	2	5
-8	-5	0	5	8
-10	-10	0	10	10
-8	-5	0	5	8
-5	-2	0	2	5

The exact nature of the kernels is not relevant to the overall system. Other edge enhancement kernels may be used in place of the above kernels.

The application of the vertical and horizontal component filters by convolver units 14 and 16 respectively results in a vertical gradient image output from convolver unit 14 and a horizontal gradient image output from convolver unit 16; the gradient image outputs are normalised in the respective convolver units and are in the form of an 8-bit value. A gradient image is an image in which the pixel values are the vertical or horizontal component of the gradient of the pixel intensity values of the original input image. The original greyscale image is differentiated in the horizontal direction to obtain the horizontal gradient image and differentiated in the vertical direction to obtain the vertical gradient image. The gradient images are then input to a polar conversion unit 18. The conversion unit 18 incorporates a look-up table (LUT) which is pre-programmed with the necessary transforms to convert from Cartesian co-ordinates to polar co-ordinates as given in Equations 1 and 2. The LUT comprises

four HM62256 RAM devices. The two component images are then combined by the conversion unit 18 on a pixel-by-pixel basis to generate two images which contain the magnitude and angle of the edges in the image. The magnitude, M, is given by:

$$M = \sqrt{x^2 + y^2} \qquad (1)$$

where x and y are the horizontal and vertical components of the edge intensities at a particular pixel.

The angle of the edges in the image (A) is given by:

$$A = \tan^{-1}\left(\frac{y}{x}\right) \qquad (2)$$

where x and y have the same meaning as for equation (1), and $\tan^{-1}$ is the arctangent function resolved over a range of $-\pi$ to $+\pi$.

The two images which contain the magnitude (M) and angle (A) of the edges in the image are output from unit 18 and input to a transputer array 20. The transputer array 20 then generates a histogram over the whole image, to provide a magnitude weighted frequency of occurrence of each possible angle value. The magnitude (M) generated by the unit 18 is normalised and output as an 8-bit binary number; the edge angle (A) generated by unit 18 is also an 8-bit binary number, giving 256 numbers to represent either the magnitude or the angle.

Rather than adding 1 to the histogram for each pixel with a particular angle, the pixel's edge magnitude is added to the histogram. This causes strong edges to make a greater contribution to the histogram than weaker ones. Very weak edges do not make a contribution, because the transputers in the array 20 impose a lower threshold on the magnitude values. This lower threshold is at a pixel edge magnitude of twenty. Pixel edge magnitudes which are less than twenty do not contribute to the histogram.

The histogram forms a 256 element vector, which describes the edge angle content of the image. The histogram "wraps around" from the angle code 255 to the angle code 0.

The transputer array 20 also smoothes the histogram by convolving it with a 25-element Gaussian kernel. This removes noise in the histogram, and compensates for the fact that the angular resolution of the histogram is limited. The values in the Gaussian kernel are as follows:

3	6	11	20	35	55	83	117	155	192	225	247	255	247	225	192	155	117	83	55	35	20	11	6	3

The 25-element Gaussian kernel is aligned with the first 25 elements of the 256-element histogram. Corresponding elements of the histogram and the kernel are multiplied together and the products summed. The resulting sum is then the smoothed histogram entry which is aligned with the centre of the kernel. The kernel is then moved along one element and the process repeated to give a second convolved histogram entry adjacent the first. This process is repeated until all 256 elements of the smoothed histogram are obtained, wrapping round when the end of the original 256 element histogram is reached, i.e. treating it as a circular structure with no start or finish.

Because the enhancement convolution kernels do not have sufficient resolution to generate all possible angle codes once the orthogonal gradient components have been converted to magnitude and angle, zeros are present in the angle histogram. These zeros introduce unwanted high angular frequency information into the histogram. The histogram is also inherently noisy due to noise in the input image; the noise being assumed to have a Gaussian distribution. The convolution with a Gaussian kernel reduces the noise and at the same time acts as a low pass filter, attenuating the high angular frequency information.

The smoothed histogram generated by the array 20 is then correlated with a reference histogram also stored in the array 20. The reference histogram is a histogram which was obtained from a previous reference image. The current

histogram is then shifted over a range of angles, centred on zero, and at each shift it is correlated with the reference histogram. The shift angle which produces the maximum correlation value is taken as the angular displacement between the current image and the reference image.

If the shift angle is not zero then there is sufficient angular deviation between the current image and the reference image to update the reference histogram. When this occurs the current histogram becomes the new reference histogram. If the shift angle is zero (all arithmetic is integer, so the shift angle is quantified into steps such that 256 steps represent 360 degrees of rotation) then the current and reference frames do not differ sufficiently to update the reference histogram. When this occurs the current reference frame remains as the reference frame. This prevents a very slow rotation in the image, i.e. a rotation where the inter-frame difference is less that 1/256 of a revolution, being missed.

The range of angles over which the correlation is performed limits the rotation speed which can be accommodated by the system 10. A larger range requires more computational effort, but results in a higher maximum rotational speed.

The angular shift with respect to the reference image is accumulated with angular shifts from previous correlations, back to some arbitrary reference time. The accumulation is performed modulo 256, to give the fraction of a complete revolution by which the current image must be rotated to align it with the original reference from the beginning of the accumulation process.

This angle is then used to rotate the current image on a pixel by pixel basis to produce an output image which is rotationally aligned with the original reference image. This is achieved by a rotator unit 22 incorporating two TMC2302 image manipulation sequencer devices using the following transform equations:

$$x' = x \cos\theta + y \sin\theta$$

and $$y' = -x \sin\theta + y \cos\theta$$

where (x, y) are the co-ordinates of a pixel in the output image and (x', y') are calculated pixel co-ordinates from where the pixel value of the input image will be taken for the pixel in the output image and θ is the rotational angle that is required to bring the image into alignment. The co-ordinates (x, y) and (x', y') are offsets from the centre of the image. Where the calculated (x', y') co-ordinates do not fall exactly on a pixel in the input image, the nearest neighbouring pixel to (x', y') in the input image is used to give the value for the output pixel at (x, y).

For square input and output images some pixels in the output image will map to pixel locations outside the input image. To produce a more aesthetically pleasing output a circular image could be produced by applying a circular mask to the output image. The diameter of the mask would equal the width or height of the image, and would only generate the pixels that fall within the mask. The image output from the rotator unit 22 is displayed on a display 24.

Figure 2 shows a further image stabilisation system 110. The system 110 incorporates components similar to the Figure 1 having reference numerals increased by 100 compared with the reference numerals of the similar components in the system 10. The system 110 includes a transputer array 130 which as well as performing the operations of the transputer array 20 of Figure 1 also performs a Kalman filter operation. A Kalman filter is an algorithm for predicting the next value of a data sequence based on analysis of the past history of the sequence and knowledge of the characteristics of a system producing the sequence. The Kalman filter is used to predict the angular displacement of the next image. This predicted angular displacement is passed to a rotator unit 122. The rotator unit 122 uses the predicted angular displacement to correct data of a new image received directly from a digitisation unit 112. The actual alignment of the new image would then be compared with the predicted alignment. The actual angular displacement would be measured and by the array 130 and used to produce an error term to improve the prediction for a subsequent image. This arrangement reduces the latency with which the system is able to display the corrected images compared with the system 10.

In an alternative system (not shown), a video signal from a camera may be routed prior to digitisation to a display device capable of displaying an image with a variable rotational displacement. Such a device might be constructed using an oscilloscope with an x-y input facility together with an intensity modulation input. The x and y inputs are driven by a digital to analogue converter controlled by address generators, constructed from programmable logic devices, which provide the necessary address sequences to give the desired rotational offset. The intensity modulation input is driven by the video signal. The desired rotational offset is determined as previously described in relation to the Figure 2 system. This system would display the corrected images with zero latency, which is of significant benefit for a control-loop system.

It is possible that a system of the invention might be subject to periodic drift in the angular rotations. This may be overcome by providing a system with a means for manually inputting an angular offset, for example by means of a thumbwheel. This manually input offset is then used to provide an initial offset to the rotation.

Whilst the previously described embodiments provide a means of stabilising an image in situations where a system camera is rotating about its optical axis, the principles behind the described image stabilisation systems may be extended to situations where an axis of rotation is parallel to but not coincident with a camera's optical axis. The algorithms required for such a stabilisation system would have to take into account the separation between the axis of rotation and the optical axis.

A component level block diagram of an image processing system 200 is shown in Figure 3. The system 200 may implement the operation of the system 10. The system 200 contains two distinct types of processor. To the left of the figure are a number of low-level pixel processing modules, referred to as digital signal processing (DSP) modules. The DSP modules are an acquisition circuit module 210, colour and monochrome display modules 212 and 214 respectively, a histogramming module 218, an FFT module 220, a robust edge detector module 222, a dual lookup table module 224 and a pipeline 226 of 2-dimensional convolvers 228. The DSP modules each have relatively limited functionality but are designed to perform common image processing tasks which are computationally very intensive such as convolution, morphology and histogramming. They are each designed to perform their specific

function with a minimum of latency from input to output, to minimise the overall latency of processor pipelines. This is important where the image processing system is part of a larger closed-loop application.

To the right of Figure 3 is a parallel array 230 of microprocessor-based image processing modules 232. High level tasks of an application are implemented in the array 230. The processing modules 232 each contain a transputer, or a combination of a transputer and a PowerPC microprocessor. The processing modules may be programmed in a number of high-level languages, such as C and OCCAM.

The processing modules 232 are interconnected by a software reconfigurable, digital video crossbar switching matrix 234. New modules may be added to the crossbar, which can accommodate up to 64 inputs and 64 outputs.

All processing modules, including the parallel processing modules, use a common video input and output protocol. This is a synchronous protocol, in which the unit of transfer is an entire image of 512 × 512 pixels, with a gap of 8 pixels between rows, to accommodate the write transfer operations in video RAM (VRAM) devices (not shown) which are used as framestores in the parallel processor array 230. A 7.5 MHz pixel clock is centrally generated, and distributed to all processors. This gives a frame transfer time of 35.5 ms, well within a nominal 40 ms frame period of the system 200. The spare time within the frame period is used to absorb the latency of DSP modules. A latency of up to 64 rows of video may be accommodated within the spare 4.5 ms, allowing very complex DSP pipelines to be constructed.

Video input is from a 25 Hz CCIR-624 monochrome source (for example, a video camera or a VCR). It is digitised to a resolution of 512 rows and 512 columns per frame, using 8 bits per pixel. There are two video displays; one is a three-colour component (RGB) display module 212, which uses a Brooktree RAMDAC to provide pseudo colour, 4-bit graphical overlay and cursor; the other display is a monochrome display module 214.

The pipeline 226 contains four convolvers 228, each capable of convolving a 3 row by 7 column kernel with an image at pixel rate, each convolver 228 being based on a

SGS Thomson A110 module. The four convolvers 228 may be operated separately, giving four different convolution operations, or may be cascaded together to give larger kernel sizes. Multiple modules may be cascaded together to increase the kernel size further.

The dual lookup table module 224 is based on a bank of SRAM, 64K by 16 bits. Two eight bit video inputs are concatenated to form the 16 bit address for the memory, and the 16 bit data produces two eight bit video outputs. By initialising the SRAM appropriately, any two functions of two input variables can be calculated in real-time. This has numerous applications such as rectangular to polar vector conversion, polar to rectangular vector conversion, arithmetic and logic operations, and thresholding operations.

The histogramming module 218 computes a grey scale histogram within up to 1024 separate regions of the input image in real-time. Histogram regions may be defined either by simple geometric partitioning of the image, or by means of a second video input (not shown) (allowing up to 256 arbitrarily shaped regions to be defined). A third video input (not shown) provides the value to accumulate to the histogram for each image pixel. This input passes through a transformation LUT which allows arbitrary weighting functions to be applied to the histograms. A microprocessor (not shown) on the module can read and analyse the histograms. For simple global histograms, the image can then be transformed using a LUT. This allows such operations as adaptive thresholding and contrast enhancement to be carried out in real-time.

The FFT module 220, based on a Sharp LH9124 FFT DSP is capable of performing a 2 dimensional FFT or IFFT on a 512 × 512 image in under 40 ms. The input and output framestores are double buffered to allow a throughput of 25 frames per second. In addition, the module contains two EPROM LUTs to enable the real time conversion between real / imaginary and magnitude / angle representation of the complex results of the transform.

The robust edge detector module 222 enhances the edges within a grey-scale image using two programmable 3 by 3 convolutions (e.g. a Sobel), then it thins the edges to

single pixel lines using non-maximal suppression. This operation is performed at pixel-rate using programmable logic hardware. The image is then hysteresis thresholded.

The processing modules are interconnected by the video crossbar switching matrix 234. This is a fully non-blocking crossbar switch, in which any input may be connected to any one or more outputs. The video crossbar switching matrix 234 may be reconfigured in software at the start of each video frame. This allows the DSP modules to be interconnected in pipelines and parallel pipelines, and allows the interconnections to be dynamically changed during algorithm execution. The video crossbar switching matrix 234 contains 64 input and 64 output ports, each of which is nine bits wide, carrying eight bits of video data and a video synchronisation signal.

The video crossbar switching matrix 234 is implemented using nine 64 by 64 single bit crossbar integrated circuits from LSI Logic, operating in a bit-sliced manner. They are configured using a transputer 236, using a set of memory mapped registers. The configuration registers are double buffered, and are clocked by the video frame sync signal. This means that the transputer has an entire frame period in which to initialise a new configuration, which then takes effect instantaneously at the next frame sync. This simplifies software, since the transputer does not have to be precise about when it changes the configuration registers.

The parallel processing array 230 consists of twenty identical image processing modules 232. Two variants of the image processing modules 232 are currently in use. The original version of the module contains a floating point (T805) transputer. The new version of the module is a hybrid of a T805 transputer with a Motorola PowerPC (either a PPC604 or a PPC603e). This is discussed in more detail below.

A block diagram of an image processing module 232 containing a transputer 250 is shown in Figure 4. The module 232 contains 4 MByte of DRAM 252, and 2 MByte of Video RAM (VRAM) 254. The VRAM 254 is used as framestore memory. It is split into two 1 MByte banks 256, each containing four 512 × 512 byte framestores. Each framestore bank 256 has a video input port, a video output port, and a map output port. Each output port has an output enable signal. The map port is a copy of the

video output port, but connects to a different bus on an array backplane, and is used when the module is operating as the array controller.

The video input/output operations are controlled by a VRAM controller 258 which is a programmable logic device. In order to perform an input or output operation, the transputer 250 writes to a control register. It is then free to continue processing with virtually continuous access to memory. The VRAM controller 258 steals a single memory cycle every four rows of video in order to perform a read or write transfer cycle within the VRAM 254. The VRAM banks 256 are independent of each other, so both banks may be transferring video data simultaneously.

The modules 232 are integrated together into the array 230 as shown in Figure 5. There are twenty identical processing modules 232 each as shown in Figure 4. The input buffer broadcasts video input to all modules using two buses IN A and IN B. The MAP OE A and B buses are output enables that determine which module will act as the controller for bus A and B. The MAP A and B buses are the controller's data which is decoded in the output decoder to enable the video output of the appropriate module using the OE A and B buses. The enabled module then outputs its data onto the OUT A or B buses, which the output buffer outputs onto the Video Out A and B buses. All transputer links go to the link crossbar switch module, which, in addition to interconnecting the links, provides eight connections into the DSP section of the system 200 architecture.

The video ports of all modules are connected to common video buses, an input and an output bus for each bank. Incoming video frames from the video crossbar switching matrix 234 are broadcast to all modules simultaneously. The software running on the array 230 determines which modules will acquire the frame. Frames are acquired in their entirety by the modules 232. The software determines which patch of the frame is to be processed by each module 232.

This approach has several benefits:

(1) it means that the algorithm may partition the processing of the frame over the array in an arbitrary manner, since all of the data is present on all processors;

(2) in non-deterministic applications which require a work farm approach to provide dynamic load balancing, the farm controller may simply pass tokens to the workers to identify patches within the frame, which minimises the communications overheads inherent in such an application;

(3) although a processor is mainly concerned with a defined patch of the image, it may occasionally require information about pixels in other parts of the image (for example at the borders of its patch), this information is available locally, so no interprocessor communication is required.

When processing of the image is completed, partial results will be contained in the framestores of various modules 232 within the array 230. If it is required to transfer the results back out of the array, for example, to display them, then it is necessary to stitch the patches together into a single coherent image. This is done as follows. One of the modules 232 is selected to be the controller for the array. This module 232 does not take part in the image processing operation; instead, it constructs within a framestore, a module address map which defines for each pixel in the result image the source module 232 for that pixel. All modules 232, including the controller, output their results simultaneously. The controller's map goes via a separate output bus, called the Map bus, to a decoder, which then generates an output enable signal which enables the output of the appropriate module. This process repeats at pixel rate, so the Output bus contains a coherent frame of data. A simplified example of this process is shown in Figure 6. The example assumes four worker modules, and shows how the map reflects the partitioning of the results across the workers.

The transputer links are connected to a crosspoint switch array. This may be configured in software, to produce any topology that can be produced with a 4-valent network. The links are generally used for high-level messages, which typically have low data rates, well below the 20 Mbit s^{-1} that the links can support. There is generally no need to pass video data between processors using the links, as the data is already broadcast to all processors.

The computational power of the T805 is modest by comparison with modern microprocessors (a 30 MHz T805 is capable of 30 MIPs and 4.3 MFLOPs). To improve the performance of the parallel processing array, the transputer modules

may be replaced by hybrid modules consisting of a T805 transputer and a PowerPC processor (either a 604 or a 603e). The modules are commercial boards from Transtech Parallel Systems Ltd., with a special interface based on VRAMs to give real-time access to the video data. The transputer is used as a communications processor, interfacing the PowerPC processor to the rest of the system in such a way that the whole module looks like a transputer, but with considerably improved performance. Code written in C for the transputer array may be ported to the PowerPC processor modules with only minor changes. A factor of between 10 and 20 speed up has been observed on a range of image processing applications comparing a 66 MHz PowerPC 604 against a 30 MHz T805.

The system 200 is housed in a 19 inch double Eurocard (6U) rack. The DSP modules have either a single or double Eurocard (220 mm by 100 or 233 mm) form factor, and connect to the video crossbar switching matrix 234 and other subsystems by means of ribbon cables with single-in-line header sockets plugging onto the back of the 96-way backplane connectors. This makes it easy to customise a system for a particular application, or to add a new module to an existing system. The parallel processing modules 232 are interconnected by a backplane, which broadcasts the video data to all modules, and implements the pixel switching technique described earlier for recombining the results from the modules.

Control of the system is via a single transputer link and transputer control signals. This may be directly connected to interface cards within a variety of hosts, including PC or SUN workstations, or by means of a transputer ethernet gateway to an ethernet network. The link has a bandwidth of 20 Mbit s^{-1}, which is sufficient for code booting, debugging and for user interaction with the system.

Algorithm development is carried out on a host computer, using cross-compilers to generate the executable code, which is then downloaded to the system 200 and run. Standard toolsets from SGS-Thomson, or from 3L support ANSI C, C++, OCCAM and FORTRAN. The DSP modules within the system 200 are viewed by the programmer as transputers which have special memory mapped peripheral devices. Framestores are viewed as two-dimensional arrays which are mapped into fixed locations in memory. Libraries of drivers for the DSP hardware have been created

and may be used by the application programmer. Code may be ported from other sources to the system 200 fairly easily, especially if the code is written in ANSI C.

Code debugging is carried out using standard debugging tools. The Inquest toolset from SGS Thomson provides full symbolic debugging facilities on all processors within the network simultaneously, and also provides network profiling tools to evaluate how efficiently the algorithm has been partitioned across the transputer array. Tools for the PowerPC offer similar facilities. An ANSI C compiler is used on the host workstation to compile object files, and a library module is called by the user's program to boot this onto the PowerPC processor at runtime. Symbolic debugging of the PowerPCs is also available, concurrently with debugging of the transputer processors.

A common problem with real-time parallel processing systems is processor synchronisation. This is handled in the system 200 as follows. Synchronisation to the video data is carried out by connecting the event (interrupt) input of all transputers in the system to the video framesync. An event handler process on each transputer can wait for the next framesync, or keep a count of framesync pulses, to synchronise the user's code. Synchronisation between processors may be carried out explicitly, by message passing between the processors, or implicitly, by waiting for framesync events.

The system has been designed as a generic architecture, for developing a wide range of real-time. It may be used for real-time correction of imagery from a rotationally unstable camera.

CLAIMS

1. An image stabilisation system comprising means for determining an angular distribution of scene edges in an acquired image and means for correlating the edge angular distribution of the acquired image with the edge angular distribution of a previous image and for generating an alignment correction in response thereto.

2. An image stabilisation system according to Claim 1 wherein the means for determining the angular distribution of scene edges comprises means for performing edge enhancement convolutions and means for generating a magnitude weighted frequency of occurrence histogram over a plurality of angles.

3. An image stabilisation system according to Claim 2 wherein the means for performing the edge enhancement convolutions comprises means for generating horizontal and vertical gradient images from the acquired image.

4. An image stabilisation means according to Claim 3 wherein the means for generating horizontal and vertical gradient images is arranged to convolve the acquired image with vertical and horizontal filter kernels.

5. An image stabilisation system according to Claim 4 wherein the horizontal and vertical filter kernels are:

5	8	10	8	5
2	5	10	5	2
0	0	0	0	0
-2	-5	-10	-5	-2
-5	-8	-10	-8	-8

and

-5	-2	0	2	5
-8	-5	0	5	8
-10	-10	0	10	10
-8	-5	0	5	8
-5	-2	0	2	5

6. An image stabilisation system according to any one of Claims 2 to 5 wherein the system includes means for performing a smoothing convolution on the histogram.

7. An image stabilisation system according to Claim 6 wherein the means for performing the smoothing convolution is arranged to convolve the histogram with a Gaussian kernel.

8. An image stabilisation system comprising means for determining an angular distribution of scene edges in an acquired image, means for correlating the edge angular distribution of the acquired image with the edge angular distribution of a previous image, means for predicting an alignment correction for the acquired image from an analysis of previous images, and means for determining an error term from a comparison of the predicted alignment correction and an actual alignment correction required to maximise the correlation between the angular distribution of the acquired image and the angular edge distribution of a previous image.

9. An image stabilisation system according to Claim 8 wherein the means for predicting the alignment correction comprises means for operating a Kalman filter algorithm.

10. An image stabilisation system according to any previous claim wherein the system includes means for enabling a system operator to correct a periodic drift in a stabilised image.

11. An image stabilisation system incorporating means for generating pixel image edge angles, wherein the rotational angle of this image is obtained from a

comparison of the pixel image edges in a current image with the pixel image edges in a reference image.

12. An image stabilisation system according to Claim 11 wherein the current image is arranged to be used as the reference image after the rotational angle is obtained.

13. An image stabilisation system according to Claim 11 or Claim 12 wherein the system includes means for enhancing spatial intensity changes in the image.

14. An image stabilisation system according to Claim 13 wherein the enhancing means incorporates separate vertical and horizontal component filters.

15. An image stabilisation system according to any one of Claims 11 to 14, wherein the representation of each detached edge angle in an image is determined by the magnitude of the pixels contributing to that edge angle.

16. An image stabilisation system according to any one of Claims 11 to 15 wherein the system incorporates a circular mask to generate a circular image.

17. An image stabilisation system according to any one of Claims 11 to 16 wherein the system includes means for predicting the angular rotation of an image, the prediction being based on the rotation of previous images.

18. An image stabilisation system according to Claim 17 wherein the predicting means includes a Kalman filter.

19. An image stabilisation system according to any one of Claims 11 to 18 wherein the system incorporates means for correcting drift in image orientation.

20. An image stabilisation system.

21. A method of stabilising an image comprising the steps of

(i) determining an angular distribution of scene edges in an acquired image;

(ii) correlating the angular distribution of the edges in the acquired image with the edge angular distribution of a previous image; and

(iii) generating an alignment correction in response to the correlation.

22. A method of stabilising an image comprising the steps of

(i) determining an angular distribution of scene edges in an acquired image;

(ii) correlating the edge angular distribution of the acquired image with the edge angular distribution of a previous image;

(iii) predicting an alignment correction for the acquired image from an analysis of previous images; and

(iv) determining an error term from a comparison of the predicted alignment correction and an actual alignment correction required to maximise the correlation between the angular distribution of the acquired image and the angular edge distribution of a previous image.

Patents Act 1977 21 Examiner's report to the Comptroller under Section 17 (The Search report)	Application number GB 9523190.8
Relevant Technical Fields (i) UK Cl (Ed.O) H4F FHHC, FESA, FESG, FESX (ii) Int Cl (Ed.6) H04N 5/232, 5/262; G02B 27/64; F41G 3/22, 7/22	Search Examiner MR J P COULES Date of completion of Search 31 JANUARY 1996
Databases (see below) (i) UK Patent Office collections of GB, EP, WO and US patent specifications. (ii) ONLINE: WPI	Documents considered relevant following a search in respect of Claims :- 1-22

Categories of documents

X: Document indicating lack of novelty or of inventive step.

Y: Document indicating lack of inventive step if combined with one or more other documents of the same category.

A: Document indicating technological background and/or state of the art.

P: Document published on or after the declared priority date but before the filing date of the present application.

E: Patent document published on or after, but with priority date earlier than, the filing date of the present application.

&: Member of the same patent family; corresponding document.

Category	Identity of document and relevant passages		Relevant to claim(s)
X	GB 2275842 A	(GOLDSTAR) see whole document	1, 8, 11, 20-22
X	GB 2074806 A	(ERLEBACH) see page 1, lines 78-86	1, 8, 11, 20-22
X	EP 0079195 A2	(CBS) see whole document	1, 8, 11, 20-22
X	WO 90/00334 A1	(PLESSEY) see whole document	1, 8, 11, 20-22

Databases: The UK Patent Office database comprises classified collections of GB, EP, WO and US patent specifications as outlined periodically in the Official Journal (Patents). The on-line databases considered for search are also listed periodically in the Official Journal (Patents).